

## Claim Amendment and Claim Listing

Please amend the claims as indicated below.

---

1. (currently amended) A method for loading and running pre-boot code on a computer from a local file, comprising the steps of:
- retrieving an image from a file on the computer into RAM, the image containing the pre-boot code;
- at least initiating redirecting I/O to emulate a peripheral storage device of the computer, whereby a subsequent call to read a sector of data from the peripheral storage device returns data from an alternate source instead of returning data from the peripheral storage device, wherein the redirecting I/O step redirects floppy I/O to read from random access memory of the computer as the alternate source; and
- reading at least a first sector of pre-boot code from the emulated peripheral storage device, and executing it, thereby passing control of the computer to the pre-boot code first sector.
2. (original) The method of claim 1, wherein the reading step reads a boot image and the passing control step passes control to an operating system which then boots on the computer, and wherein the computer boots from a different operating system when the redirecting step is not used.
3. (original) The method of claim 2, wherein the boot image includes DOS operating system code, and the passing control step results in booting DOS on the computer.
4. (original) The method of claim 1, wherein the reading step reads an image containing pre-boot code which flashes BIOS in a memory of the computer.

5. (original) The method of claim 1, wherein the reading step reads an image containing pre-boot code which logs a subsequent boot process to detect boot failure and assist determination of a cause of boot failure on the computer.

6. (canceled).

7. (currently amended) The method of claim [6] 1, wherein the peripheral storage device is a floppy drive and the method comprises loading a floppy image into extended memory of the computer as the alternate source.

8. (currently amended) The method of claim [6] 1, wherein the method comprises detecting the presence of a memory manager and loading a floppy image into memory allocated by the memory manager before code other than the pre-boot code allocates memory using the memory manager.

9. (original) The method of claim 1, wherein the retrieving step comprises reading pre-boot code from the file into a contiguous region of random access memory in the computer, and the redirected I/O reads pre-boot code from such a contiguous region as the alternate source.

10. (original) The method of claim 1, wherein the retrieving step comprises reading pre-boot code from the file into at least one region of random access memory in the computer, and the redirected I/O reads pre-boot code from the random access memory as the alternate source.

11. (currently amended) The method of claim 1, wherein the retrieving ~~reading~~ step comprises reading pre-boot code from the file directly for execution, sector by sector, using redirected I/O to read the file as the alternate source.

12. (original) The method of claim 11, wherein at least two sectors of the file are not contiguous on disk.

13. (original) The method of claim 1, wherein the method comprises setting a default item in a Windows boot.ini file.

14. (original) The method of claim 1, wherein the method comprises changing the name of a file an NT boot record will load.

15. (original) The method of claim 1, wherein the method comprises substituting other pre-boot code for standard loader code.

16. (original) The method of claim 1, wherein the method comprises using the default boot.ini entry to identify non-standard pre-boot code to load.

17. (original) A computer system configured to run pre-boot code from a local file, the computer system comprising:

a processor in operable connection with a random access memory;

a local hard drive accessible to the processor without a network connection, the local hard drive storing a partition, the partition containing a file system, the file system containing a file, the file containing the pre-boot code; and

a pre-boot code loading means for reading the pre-boot code from the file into the random access memory for execution by the processor prior to or in place of booting on the computer system an operating system which is distinct from the pre-boot code.

18. (original) The configured computer system of claim 17, wherein the pre-boot code loading means comprises code for redirecting floppy drive I/O to read pre-boot code from a contiguous region in the random access memory.

19. (original) The configured computer system of claim 17, wherein the pre-boot code loading means comprises code for redirecting floppy drive I/O to read pre-boot code from at least two non-contiguous regions in the random access memory.

20. (original) The configured computer system of claim 17, wherein the pre-boot code loading means comprises code for redirecting floppy drive I/O to read pre-boot code from the file without requiring a booted file system and a booted operating system.

21. (original) The configured computer system of claim 17, wherein the pre-boot code loading means identifies the pre-boot code file by using a default item in a boot.ini file.

22. (original) The configured computer system of claim 17, wherein the pre-boot code loading means changes the name of a file an NT boot record will load.

23. (original) The configured computer system of claim 17, wherein the pre-boot code loading means substitutes pre-boot code for standard NT loader code.

24. (original) The configured computer system of claim 17, wherein the pre-boot code loading means reads a sector data structure that identifies sectors of the file that contains the pre-boot code.

25. (original) The configured computer system of claim 17, wherein the pre-boot code loading means calls an extended memory manager to allocate a new region of extended memory, and copies pre-boot code into that new region

26. (original) The configured computer system of claim 17, wherein the pre-boot code loading means caches pre-boot code, whereby part of the pre-boot code is in memory and part is on disk.

27. (original) The configured computer system of claim 17, wherein the pre-boot code loading means permits the system to be booted into a DOS operating system without using a boot floppy disk and without booting off of a DOS hard disk partition.

28. (original) A configured computer program storage medium which contains software to perform method steps for running pre-boot code from a local file, the method steps comprising the steps of:

redirecting I/O of a peripheral storage device of the computer, whereby a subsequent call to read a sector of data from the peripheral storage device returns data from an alternate source instead of returning data from the peripheral storage device; obtaining an image from a file on the computer, the image containing the pre-boot code, the image obtained at least in part by using redirected I/O; and passing control of the computer to the pre-boot code.

29. (original) The configured computer program storage medium of claim 28, wherein the obtaining step obtains a boot image and the passing control step passes control to an operating system which then boots on the computer.

30. (original) The configured computer program storage medium of claim 29, wherein the boot image includes DOS operating system code, and the passing control step results in booting a DOS operating system on the computer without using a boot floppy disk and without booting off of a DOS hard disk partition.

31. (original) The configured computer program storage medium of claim 28, wherein the redirecting I/O step redirects floppy I/O to read from a random access memory of the computer as the alternate source.